ROS in Autonomous Systems Applications
(Raspberry Pi / ROS Installation - Process)

This Document is prepared to help in the installation process of ROS Melodic on a Raspberry Pi Model 3 B+

Table of Contents

List of Components 2

How to write the Raspberry Pi OS on the SD card 3

The installation process for the *Raspberry Pi OS* from *Windows*: 3

Install ROS Melodic on Raspberry Pi 4

The installation process for the *ROS Melodic* on *Raspberry Pi OS*: 4

©Multi-Robot Systems (MRS) Research Group

ROS in Autonomous Systems Applications
(Raspberry Pi / ROS Installation - Process)

List of Components

Item No.	Item	Specifications
1	SD Card and Reader	• Minimum size 16Gb for the full OS Version; class 4 (the class indicates how fast the card is). • We recommend using branded SD cards as they are more reliable.
2	HDMI to HDMI	To connect the raspberry pi with external screen (laptop screens do not work)
3	Keyboard and Mouse	USB keyboard and mouse to be used while navigating the OS. Preferrable wireless with a single USB dongle to minimize the used USB ports
4	Ethernet Cable	To connect the Pi with the internet (Wireless communication is also available on the Pi)
5	Power Adaptor	Use a good stable power adaptor with micro USB cable

©Multi-Robot Systems (MRS) Research Group

ROS in Autonomous Systems Applications
(Raspberry Pi / ROS Installation - Process)

How to write the Raspberry Pi OS on the SD card

- There are several ways to install the Raspberry Pi OS (previously called Raspbian) on the SD card.
- You can install previous version of Raspbian, Noobs, or you can install the OS from the Raspberry Pi Imager.
- Below, you can find the steps to follow on windows in order to install the Raspberry Pi OS using imager.

The installation process for the *Raspberry Pi OS* from *Windows*:

1. Download the Raspberry Imager from the below link and install it https://downloads.raspberrypi.org/imager/imager_1.5.exe
2. Connect an SD card reader with the SD card inside.
3. Open Raspberry Pi Imager and choose the required OS from the list presented.
4. Choose the SD card you wish to write your image to.
5. Review your selections and click **WRITE** to begin writing data to the SD card.
6. It will take a while till finalizing both writing and verifying stages then the imager will ask you to remove the SD Card.
7. Now, the Raspbian is written on your SD Card.
8. Remove the SD Card and insert it in your Raspberry Pi.

Note: In case windows 10 is with Controlled folder access enabled, make sure to give access to the imager otherwise the writing process would fail.

1. From Start Menu > Settings > Update & Security > Windows Security > Virus & threat protection.
2. Under Virus & threat protection settings, select Manage settings.
3. Under Controlled folder access, select Manage Controlled folder access.
4. Switch the Controlled folder access setting select allow an app through Controlled folder access.
5. Press add an allowed app > Browse all apps > select the imager.

©Multi-Robot Systems (MRS) Research Group

ROS in Autonomous Systems Applications
(Raspberry Pi / ROS Installation - Process)

Install ROS Melodic on Raspberry Pi

- The ROS melodic is currently the most stable version of ROS that can be installed.
- Most of this section referred to a tutorial in ROS wiki If you don't understand this instruction or have any problem, see the tutorial. http://wiki.ros.org/ROSberryPi/Installing%20ROS%20Melodic%20on%20the%20Raspberry%20Pi

The installation process for the *ROS Melodic* on *Raspberry Pi OS*:

These instructions assume that Raspbian Buster is being used as the OS on the Raspberry Pi.

1. **Setup ROS Repositories**
   a. `sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'`
   b. `sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654`
2. **Update your Debian package index**
   a. `sudo apt-get update`
   b. `sudo apt-get upgrade`
3. **Install Bootstrap Dependencies**
   a. `sudo apt install -y python-rosdep python-rosinstall-generator python-wstool python-rosinstall build-essential cmake`
4. **Initializing rosdep**
   a. `sudo rosdep init`
   b. `rosdep update`
5. **Create a catkin Workspace**
   a. `mkdir -p ~/catkin_ws_pi`
   b. `cd ~/catkin_ws_pi`
6. **Install the ROS-Comm (*No gui)***
   a. `rosinstall_generator ros_comm --rosdistro melodic --deps --wet-only --tar > melodic-ros_comm-wet.rosinstall`
   b. `wstool init src melodic-ros_comm-wet.rosinstall`

   *In case of any failure or interruption, try to reduce the system parallel computing by resuming using -j4 instead of -j8 by running the command (*The -jx option downloads x packages in parallel*)*
   a. `wstool update -j4 -t src`
7. **Resolving Dependencies with rosdep**
   a. `cd ~/catkin_ws_pi`

©Multi-Robot Systems (MRS) Research Group

b. `rosdep install -y --from-paths src --ignore-src --rosdistro melodic -r --os=debian:buster`

8. **Building the catkin Workspace**
   a. `sudo ./src/catkin/bin/catkin_make_isolated --install -DCMAKE_BUILD_TYPE=Release --install-space /opt/ros/melodic`

   The raspberry pi might **freeze** while compiling, this is due to the high computations. To solve that:
   - Restart your raspberry pi
   - Reduce the computation by adding -j2 or even -j1 option
     `sudo ./src/catkin/bin/catkin_make_isolated --install -DCMAKE_BUILD_TYPE=Release --install-space /opt/ros/kinetic -j2`

9. In order to build the ROS workspace every time you make a change or add a package, you have to run the 3 following commands to avoid any errors in building the workspace.
   1. `source /opt/ros/melodic/setup.bash`
   2. `cd ~/catkin_ws_pi`
   3. `catkin_make`
   4. `source devel/setup.bash`
10. **Now ROS should be installed!**
    a. Test the installation by running the command `roscore`
11. In order to avoid calling the 3 commands continuously, you can add command (**a** and **c**)
    a. `cd ~`
    b. `gedit .bashrc`
       (*if gedit is not installed, you can use* `sudo apt-get install gedit`)
    c. Add the below lines at the end of the file and save.
       i. `source /opt/ros/melodic/setup.bash`
       ii. `source ~/catkin_ws_pi/devel/setup.bash`
    d. Now, you can only build the workspace using `catkin_make`
12. Now, you can start the beginning ROS tutorials
    http://wiki.ros.org/ROS/Tutorials

©Multi-Robot Systems (MRS) Research Group